Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 10 (2015) 554 – 563

Transportation Research Procedia

www.elsevier.com/locate/procedia

18th Euro Working Group on Transportation, EWGT 2015, 14-16 July 2015, Delft, The Netherlands

A new approach to crew scheduling in rapid transit networks

Manuel Fuentes [a,*], Luis Cadarso [a], Ángel Marín [b]

[a] Rey Juan Carlos University, Departamental III Building Camino del Molino s/n, Fuenlabrada 28943, Spain
[b] Technical University of Madrid, Pza. Cardenal Cisneros 3, Madrid 28040, Spain

---

Abstract

We propose a new approach for the crew scheduling problem in rapid transit networks. With this approach we try to open a new branch for future research, providing a different way of facing the crew scheduling problem which makes integration with other planning problems easier than the traditional approach based on column generation for solving a set covering/partitioning problem. For solving this new model we develop a Lagrangian relaxation and we take advantage of an ad hoc decomposition based on time-personnel clustering. We present some preliminary computational experiments for real case studies drawn from the main Spanish train operator, RENFE.

Peer-review under responsibility of Delft University of Technology

Keywords:
Crew scheduling, Lagrangian relaxation, Rapid transit networks.

---

1. Introduction

Planning processes related to railway systems are fields that are rich in combinatorial optimization problems. Due to the tremendous size of real instances, the planning process is usually divided into several steps such as network design, line planning, timetabling, rolling stock circulations and crew planning (Huisman et al. (2005), Cadarso and Marín (2014)):

1. Network design: designing a Rapid Transit Network (RTN) is a vital strategic subject due to the fact that it reduces the future traffic congestion, travel time and pollution. The location decisions and the maximum coverage of the demand for the new network is the main goal.
2. Line planning: the following step after designing a RTN is planning its lines (origin and destination stations, stops and frequencies). The problem of designing a line system aims at satisfying the travel demand while maximizing the service towards the passengers or minimizing the operating costs of the railway system.

* Corresponding author. Tel.: +34-914888759; fax: +34-914887500.
E-mail address: manuel.fuentes@urjc.es

Peer-review under responsibility of Delft University of Technology
doi:10.1016/j.trpro.2015.09.009

3. Timetabling: the general aim of the railway timetabling problem is to construct a train schedule that matches the frequencies determined in the line planning problem.
4. Rolling stock circulations: they are determined in order to satisfy both the timetable and the demand. The train unit type and composition assignment to each train service and a sequence for each train unit in the network are determined.
5. Crew planning: it consists in assigning different tasks (e.g., train services) that must be done to available workers (the crewmembers). While the idea is simple, the problem can get very complex due to different matters (e.g., strong labor rules). This and the complexity of real world instances make the crew planning problem a challenge, so it is usually split into two different phases: the crew scheduling and the crew rostering problems. The former one focuses on a short planning period while the latter one focuses on long planning periods.

In the last few decades, the Crew Scheduling Problem (CSP) has been studied widely. The increase in research attention could be motivated by economic considerations. For many companies, labor cost is the major direct cost component. Cutting this cost by only a few percent by implementing a new crew schedule could therefore be very beneficial (Van den Bergh et al. (2013)).

The work presented here is also focused on the CSP, which consists in assigning tasks to generic crews in a short temporal horizon. Tasks are usually grouped into duties. We assume that the CSP is tackled just after rolling stock circulations have been determined.

Almost all the research made before related to the CSP follows the same philosophy: the problem is divided into two different problems, making use of the column generation approach (Barnhart et al. (1998)). One of the problems consists in a set partitioning (or set covering, depending on how deadheads are considered) where a subset of duties covering all the tasks at a minimum cost is determined. The other one, the duty generation problem, consists in generating feasible candidate duties (columns). Duties are usually generated using constrained shortest path algorithms and taking into account dual prices of the master problem. The solution finding process is therefore iterative, alternating between both problems until the optimal (or a good enough solution) is found.

The classical point of view, which has been extensively applied to airlines and railways (Barnhart et al. (2003),Kroon and Fischetti (2001)), offers good advantages such as being able to deal with non-linear payment systems or very complex labor rules. However, in this paper we present a new approach to deal with the CSP in RTNs; such networks operate in metropolitan areas, and feature frequent train services and heavy passenger loads. The new approach presented here offers the following advantages, which exploit the particularities of RTNs such as their linear payment systems. First, the integration of the CSP with other planning problems becomes easier, since it is based on sequencing. Second, its application to recovery problems may exploit the fact that previously computed duties which are affected by a disruption are not forced to be discarded totally, but only partially. And finally, it opens a new branch for future research, providing a different way of facing the CSP.

This paper is organized as follows. In Section 2 a literature overview and paper contributions are given. Section 3 describes the problem in detail. Section 4 is devoted to the mathematical model. In Section 5 we present our solution approach. In Section 6 we show the computational experiments we have performed. Finally, we draw some conclusions in Section 7.

2. Literature review

The CSP has been widely studied in the literature. Van den Bergh et al. (2013) presents a survey for the personnel scheduling, which covers, among others, the CSP applied to different transport modes.

Concerning the airline industry, a industry where a great research effort has been done in the CSP, Barnhart et al. (2003) describe the crew planning problem, referred as crew scheduling, which is divided into crew pairing and crew rostering problems (nomenclature in the literature is different depending on the transport mode under study). They model both problems as set partitioning or set covering problems, which represent the most common formulations. Bomdörfer (1998) investigates polyhedral, algorithmic and application aspects of such models.

For railway applications, Huisman et al. (2005) present a survey of the general planning process. Kumar et al. (2009) describe the railroad crew scheduling problem defining terminology and labor rules. They tackle the problem as an integer programming model and present several solution methods. Caprara et al. (1997) give an outline of different ways of modeling the crew scheduling and rostering problems and possible solution methods. Two main solution approaches are illustrated for real-world applications. Kroon and Fischetti (2001) study the CSP making use of a set covering model with additional constraints. They solve real case studies drawn from the Dutch railway operator NS Reizigers using dynamic column generation techniques, Lagrangian relaxation and heuristics. Abbink et al. (2005) presented an application of an operations research model to develop an alternative set of scheduling rules. This alternative set of rules satisfied the crewmembers' requests and improved the railway operator's punctuality and efficiency also cutting personnel costs.

Contributions

The main contributions of this work are in the model presented and in the solution approach, which are shown to be rather effective from a computational point of view.

First, we develop a new mathematical formulation which generates and selects the duties to be operated in a rapid transit network. This approach is able to solve the CSP in a reasonable amount of time making use of sequencing. Sequences are not directly given in the definition of the variables, which is the traditional approach for set covering and partitioning problems. This approach is better from the point of view of a possible integration with previous planning stages. When solving previous planning problems, such as the rolling stock assignment problem, tasks to be operated are variables to be decided (e.g., composition changes). Traditional formulations need to know the tasks to construct the columns of the model while a sequencing model may easily include this issue.

Second, we propose a solution approach which makes use of an ad hoc decomposition based on time-personnel clustering, exploiting some properties of the network. In order to find optimal or near-optimal solutions we make use of a heuristic and of the Lagrangian Relaxation. The heuristic finds feasible solutions to the problem while the Lagrangian relaxation computes lower bounds of the optimal solution in order to demonstrate that the solutions given by the heuristic are optimal or near-optimal solutions.

3. Problem description

The CSP is an assignment problem. It consists in assigning different tasks to certain crewmembers. The rest of this section describes the problem in detail. First, the network and schedule are introduced. Next, available resources are detailed. Finally, duties and problem objectives are introduced.

3.1. Network and schedule

The railway network consists of tracks and stations. Depot stations form a subset of the stations; these are the locations where trains are parked or shunted, and where crew bases are located. Between two stations, two different tracks exist, one for each direction of movement.

Train services are grouped in lines. A line is characterized by its terminal stations, by a path through the infrastructure between the terminals, and by a set of stations along the path. Train services run up and down between the terminals and call at the specified stations underway. The timetable departure times and frequencies are fixed and publicly available. RTNs are characterized by high frequencies and a lack of capacity in depot stations. These facts make it difficult to operate the network without empty services. These are defined by an origin, a destination and a departure time. Empty services can help satisfy both capacity and train unit availability in depot stations.

3.2. Resources

A RTN operator has two main resources to be scheduled: rolling stock and crewmembers.

The rolling stock is composed of self-propelled train units which can be of different types. All of them have a driver seat at both ends. Units of the same type can be attached to each other to form trains compositions. A composition of train units is a sequence of elements of the same type. Although composition changes enable the network operator to use smaller fleet sizes, it is always a complicating operation, due to the necessity of human resources.

Crewmembers, in general railway applications, may be mainly classified into drivers (the ones driving the train) and conductors (the ones responsible for operational and safety duties that do not involve actual operation of the train).

In the context of RTNs, conductors are seldom needed. Moreover, since labor rules applying to drivers are generally stricter, there is no loss of generality if only drivers are considered; drivers and crewmembers (or just crews) are equivalent for the remainder of the paper. However, the model presented here can be extended to conductors easily.

These two types of resources, namely rolling stock and crewmembers, are assigned to timetabled services in order to get fully operational schedules. Rolling stock circulations are usually solved before crew scheduling. And in order to schedule crewmembers the following rules must be considered.

Crew rules

The rules and regulations express the conditions under which a crew assignment is considered legal. Some of these might be due to legislation and governmental agencies, some might be imposed by the railway operator itself, and others are due to agreements between the company and the employee unions.

Because we are dealing with the CSP in RTNs, where the planning horizon is usually one day, we consider the following crew rules:

- Crew, task, and time compatibility rules check the legality of their respective assignation. Thus, incompatibility may arise due to lack of crew qualifications, needs or pre-assignments.
- Rest time between tasks determines whether two tasks can follow immediately after each other.
- Accumulated rules limit the number of working hours.

Due to the small size of RTNs (in the sense of spatial extension), no overnight rests out of base are considered; this fact simplifies the problem. On one hand, costs structures are simpler. On the other hand the problem can be solved for separate days.

3.3. Duties and objectives

All the information described above in this section (i.e., schedule, rolling stock circulations and crew availability) becomes readily available when facing the CSP. From the point of view of crew scheduling and without loss of generality, we define the set of tasks as the set composed of the following elements: train services, empty services and composition changes.

Duties are sequences of tasks. Each duty must be performed by a unique crewmember. Duties usually span one working day (i.e., work shift) within RTNs. However, they can be shorter if split work shifts are considered. Anyway, every duty must comply with rules and regulations specified above. Obviously, apart from these rules and regulations, when constructing feasible duties physical constraints must be also satisfied. That is, the tasks must be sequential spatially: each task must end at the same location at which the subsequent task is starting.

There are different objectives a RTN operator may want to accomplish: minimization of the cost of assigning crews to their respective tasks, minimization of, and often avoid, unassigned tasks, and minimization of overtime payment. Any of those potential objectives can be considered in the model we present, but we perform our tests focusing on the first one.

4. Mathematical model

We consider RTNs, where the planning period is usually reduced to a single day, and duties are at most one work shift long. Moreover, particularities of RTNs, where frequencies are usually high and movements short, lead to simpler payment systems. The previous motivates us to develop this alternative formulation, which is based on sequencing. We aim at developing a model which could be integrated easily with other planning stages too and extended in order to be used for different cases, even for the crew rostering problem.

The SEquencing BAsed Crew Scheduling (SEBACS) model is an Integer Linear-Programming (ILP) model. The mathematical formulation is based on a network where the nodes are the tasks to be covered. Arcs are therefore connections between them. There are two basic kinds of arcs: direct connection arcs and extreme connection arcs. The former refer to connections between two consecutive tasks, while the latter characterizes duties by means of their first and last tasks.

In order to be able to formulate the SEBACS model, we need to define the following sets and parameters:

- M: set of tasks indexed by i,j and k. Each task is defined by its origin and destination stations and its departure and arrival times;
- D: set of potential duties indexed by d. Its cardinality is the maximum number of available drivers;

- $c^{i,j} \in \{0, 1\}$: parameter defined for $i, j \in M$ to indicate whether tasks i, j can be consecutive in a duty (i.e., connected by a direct connection arc);
- $e^{i,j} \in \{0, 1\}$: parameter defined for $i, j \in M$ to indicate whether tasks i, j can be the first and last tasks (i.e., extreme tasks) respectively in a duty;
- $a^{i,j} \in \mathcal{R}$: parameter defined for $i, j \in M$ to indicate the cost of a duty which has tasks i, j as extreme tasks;
- $b^i \in \mathcal{R}$: parameter defined for $i \in M$ to indicate the cost of not covering task i;

Main decision variables are those stating whether a task is assigned to a certain duty or not, and those indicating if an arc (of any kind) is included in a duty as a part of the solution. These are:

- $\alpha_i^d \in \{0, 1\}$, defined for $i \in M, d \in D$, to indicate whether task $i \in M$ is done within duty $d \in D$;
- $\gamma_{i,j}^d \in \{0, 1\}$, defined for $i, j \in M, d \in D$, to indicate whether tasks $i, j \in M$ are consecutive (i.e., connected by a direct connection arc) within duty $d \in D$;
- $\epsilon_{i,j}^d \in \{0, 1\}$, defined for $i, j \in M, d \in D$, to indicate whether tasks $i, j \in M$ are the first and last tasks (i.e., connected by an extreme connection arc) within duty $d \in D$, respectively.

The objective function is as follows:

$$\min \sum_{i\in M} \sum_{j\in M} \sum_{d\in D} a^{i,j} e^{i,j} \epsilon_{i,j}^d + \sum_{i\in M} b^i (1 - \sum_{d\in D} \alpha_i^d) \tag{1}$$

The objective function in (1) consists of two terms: a term depending on the number of duties present in the solution and another term corresponding to uncovered tasks. The former one determines the total cost of the duties present in the solution while the latter one penalizes the tasks which remain uncovered. The reason why we are considering uncovered tasks is that it improves the solution finding process with the heuristic we are proposing in the next section. With the appropriate cost associated to uncovered tasks, they can be considered as covered by a duty composed of just one task.

The core constraints are as follows:

$$\sum_{d\in D} \alpha_i^d \leq 1 \qquad \forall i \in M \tag{2}$$

$$\sum_{i\in M} \sum_{j\in M} e^{i,j} \epsilon_{i,j}^d \leq 1 \qquad \forall d \in D \tag{3}$$

$$\sum_{i\in M} c^{i,j} \gamma_{i,j}^d + \sum_{k\in M} e^{j,k} \epsilon_{j,k}^d = \alpha_j^d \qquad \forall j \in M, d \in D \tag{4}$$

$$\sum_{j\in M} c^{i,j} \gamma_{i,j}^d + \sum_{k\in M} e^{k,i} \epsilon_{k,i}^d = \alpha_i^d \qquad \forall i \in M, d \in D \tag{5}$$

Constraints (2) state that each task i is included within one duty d at most, but it may remain uncovered. Constraints (3) ensure that there is only one pair of extreme tasks per duty at most. That is, if a duty d is used in the solution, it will only contain one first task and only one last task. Constraints (4) and (5) are sequencing constraints. The former ones determine that a specific task in a duty must be followed by just another task. This is, if a task j is being performed in a duty, then it must be immediately preceded by one task i. If it is not preceded by any other task, it means it is the first task in the duty. A similar idea applies to constraints (5), but for the succeeding tasks.

Rest (i.e., drivers' continuous working time is limited) and deadheading (i.e., drivers do not always have to drive to go from one station to another) constraints are not included here due to space limitations. We model deadheading with a binary variable δ_i^d which takes value 1 if task i belongs to duty d as deadhead; then, these variable are added to constraints (4) and (5) with additional indicator parameters.

Limitation of driving time within a duty can be added through constraints such as $\sum_{i \in M} dt^i \alpha_i^d \leq maxt_d$, where dt^i is the driving time of task i and $maxt_d$ is the maximum driving time for duty d. Crew qualifications may be easily included in our model formulation; we did not include them because the our case study has only one rolling stock type and all the tasks are similar. In order to include crew qualifications, we need to split the set of duties D into different subsets, a subset for each different crewmember. Then, an indicator parameter would define whether a task is compatible with a crewmember.

Note that no constraints related to crew bases or duty time are needed, since information about them, such as crew base compatibility and maximum duty time, is implicit in $e^{i,j}$. Moreover, the nature of rapid transit networks make possible to avoid overnight rests; this is the ground truth for our case study.

5. Solution approach

Due to combinatorial complexity (there are millions of binary variables in a typical real instance), the solving process becomes very hard. Exact methods and/or commercial software fail to solve real case studies. In order to efficiently solve the problem proposed here, we use the concept of clustering. Then, a heuristic is used to obtain feasible solutions to the problem and Lagrangian relaxation to obtain lower bounds to the optimal solution.

5.1. Clustering

Due to the nature of the network and the mathematical formulation proposed, the model size can be reduced by designing a personnel-temporal (i.e., duty-temporal) based clustering.

Tasks span a whole schedule (i.e., the day under study), which covers a time period much larger than the maximum duty time allowed by labor rules (almost three times in RTN). This fact makes impossible to cover the following two tasks in the same duty: one task in the beginning of the considered schedule and another one in the end in the same schedule.

The clustering we have developed consists in creating time clusters spanning the whole schedule over which we distribute all the potential duties. Each potential duty is included in one and only one cluster. The clusters must cover an enough long time which must be greater than the maximum duty time. There is overlapping between clusters in terms of time and therefore tasks; this is, tasks are usually included in several clusters. We denote clusters by $\ell \in L$. M_ℓ and D_ℓ are the subset of tasks and duties within cluster ℓ, respectively. Note that D_ℓ is a partition of D, while M_ℓ is not a partition of M, but their union is M ($D = \bigcup_{\ell \in L} D_\ell$ and $\bigcap_{\ell \in L} D_\ell = \emptyset$; $M = \bigcup_{\ell \in L} M_\ell$ and $\bigcap_{\ell \in L} M_\ell \neq \emptyset$).

The number and length of clusters is estimated empirically. The time allocated to each of them is the maximum duty time increased by a factor greater than one. Clusters may overlap; the time separation between the starting time of two consecutive clusters is determined as a fraction of the maximum duty time. The number of clusters is determined using the length and the time separation of the clusters, forcing the whole day to be covered by at least one cluster. The number of potential duties assigned to a cluster is computed taking into account two parameters: the number of tasks present on the cluster, and the number of times each task is present in different clusters. This clustering, when the involved parameters are chosen carefully, produces exactly the same solution than the original model without clusters; this is, there is no loss of generality at all due to it.

5.2. Heuristic

However, the clustered-SEBACS is still non-tractable by exact methods. But we are able to obtain feasible solutions using a heuristic such as the Relaxation Induced Neighborhood Search (RINS) Danna et al. (2005). RINS is a heuristic that explores a neighborhood of the current incumbent solution to try to find a new, improved incumbent. At a node of the global branch-and-cut tree, the following steps are performed:

1. first, fix the variables that have the same values in the incumbent and in the current continuous relaxation;
2. second, set an objective cutoff based on the objective value of the current incumbent;
3. finally, solve a sub-ILP model on the remaining variables. The RINS sub-ILP model is also potentially large and difficult to solve, so its exploration must often be truncated.

5.3. Lagrangian relaxation

However, optimality cannot be proven with the RINS. We need to obtain lower bounds to the optimal solution in order to determine how far the solutions obtained by the RINS are from the optimum.

Clustering makes the mathematical model separable into smaller, simpler and independent sub-problems: one sub-problem for each cluster. This fact makes the Lagrangian relaxation (Fisher (2004)) suitable for the purpose of obtaining good-quality lower bounds to the optimal solution. We relax constraints (2). Therefore, the objective function for each cluster ℓ in the Lagrangian relaxation approach is as follows:

$$\min \sum_{i\in M_\ell}\sum_{j\in M_\ell}\sum_{d\in D_\ell} a^{i,j}e^{i,j}\epsilon^d_{i,j} + \sum_{i\in M_\ell} b^i(1-\sum_{d\in D_\ell}\alpha^d_i) + \sum_{i\in M_\ell}\sum_{d\in D_\ell}\mu_i\alpha^d_i \tag{6}$$

where μ_i are the dual variables of constraints (2).

When we solve the submodels (one for each cluster ℓ) using the optimal values of μ_i, we obtain the solution given by the mathematical model in (1)-(5). Now, the problem is to determine the optimal values for μ_i. Subgradient and Ellipsoid methods could be used for that purpose. However, these iterative methods are simple and usually fail to converge to the optimal values of the dual variables (see Lemaréchal (2001)).

With slight abuse of notation, the dual function associated with (6) is the function of μ defined by

$$\mathcal{R}^+ \ni \mu \to \theta(\mu) := \min_{\epsilon,\alpha,\gamma\in\chi} \mathcal{L}(\epsilon,\alpha,\gamma,\mu). \tag{7}$$

The dual problem is then

$$\max \theta(\mu), \mu \in \mathcal{R}^+. \tag{8}$$

For given values of μ, say μ_{it}, we are able to solve problem (7) easily. Each solution defines an affine function approximating θ from above: $\theta(\mu) \leq \theta(\mu_{it}) + g_{it}(\mu - \mu_{it})$ for all $\mu \in \mathcal{R}^+$, where g is a subgradient of θ, which elements are $g_i = \sum_{d\in D}\alpha^d_i - 1$.

After several calls (i.e., solutions to (7) once μ is known), say IT, we have a piecewise function $\hat{\theta}$ which overestimates θ: for all $\mu \in \mathcal{R}^+$,

$$\theta(\mu) \leq \hat{\theta}(\mu) := \max\left\{\theta(\mu_{it}) + g^{it,T}(\mu-\mu_{it}) : it = 1,...,IT\right\}, \tag{9}$$

which is called the cutting-plane model of θ.

Assuming that the model approximates correctly the true θ, maximizing it makes sense; and this is a linear programming problem. Thus, the method consists in computing an optimal solution (μ_{it+1}, r_{it+1}) of

$$\max r \tag{10}$$

$$r \leq \theta(\mu_{it}) + g^{it,T}(\mu-\mu_{it}) \qquad \forall it \in IT \tag{11}$$

$$r \in \mathcal{R} \tag{12}$$

$$\mu \in \mathcal{R}^+. \tag{13}$$

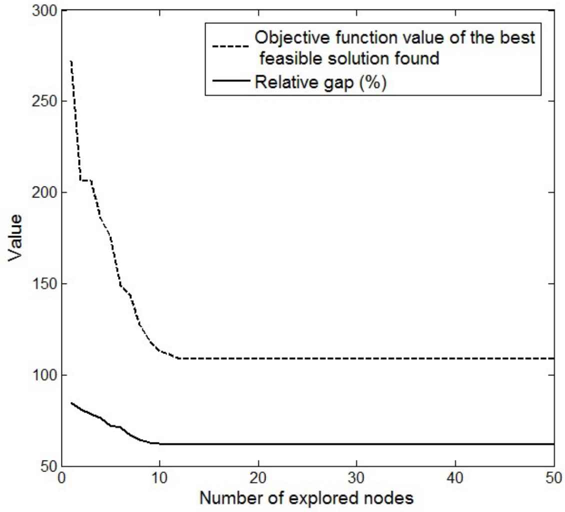

Fig. 1. Objective function and optimality relative gap values provided by the RINS vs. nodes of the branch-and-cut tree.

6. Computational experiments

Our preliminary experiments are based on simplified cases drawn from RENFE's regional network in Madrid, Spain. This network is composed of 10 different lines with almost 100 stations, carrying more than one million passengers every day. Our case study is drawn from Line C5, which has 23 stations and 4 depot stations. We focus on Line C5 because is the most dense line in the network regarding number of tasks. It has more than 430 tasks on a typical day of operations. This line features one rolling stock type and there are 72 available train units. For the preliminary computational experiments in this paper, we included the 35% of a typical day of operations. We use as input data the schedule provided in Cadarso and Marín (2010) and Cadarso and Marín (2011).

We used for our tests a laptop with an Intel Core i7 at 2 GHz and 8 GB of RAM, running under Windows 8.1 64-Bit, and we implemented the models in GAMS/Cplex 12.1.

Table 1 shows the mathematical model size. The clustered-SEBACS and SEBACS models numbers of discrete variables, constraints and non-zero elements are given. As we can see, the model size is greatly reduced without sacrificing the optimal solution.

Table 1. SEBACS model size.

Item	Clustered-SEBACS	SEBACS
Number of discrete variables	16798	30303
Number of constraints	10745	16389
Number of non-zero elements	75064	144375

Figure 1 shows the objective function and the optimality relative gap values provided by the RINS vs. nodes of the branch-and-cut tree and computational time. The RINS is able to improve the solution during the first nodes: the objective function value is lowered. However, this approach stalls and it cannot prove optimality as shown in Figure 1; the optimality relative gap reaches a value of approximately 60% and remains constant. However, this does not necessarily mean we have not found the optimum. Figure 2 shows the lower bounds obtained by the Lagrangian relaxation of the model at each iteration. The lower bound increases until the iterative approach terminates at a value of approximately 108. Comparing this lower bound with the lowest objective function value provided by the RINS, which is showed in Figure 1, we see that the RINS has effectively found the optimal solution.

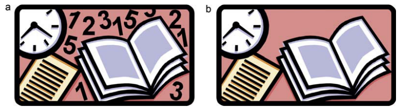

Fig. 2. Lower bounds obtained by the Lagrangian relaxation of the model at each iteration.

Table 2 shows the clustered-SEBACS model solutions provided by the RINS for different nodes of the branch-and-cut tree. The first column displays the node number. The second one the number of duties in the solution. The third column shows the efficiency of the solution. And the last one the objective function value. The efficiency of the solution is defined as the average efficiency of all duties in the solution (considering each unassigned task as a whole duty). The efficiency of a duty is defined as its actual driving time divided by its theoretical maximum driving time. This theoretical maximum time is defined as the maximum duty time minus the minimum break time. This parameter helps comparing solutions. As we have demonstrated above, the last row of Table 2 contains the optimal solution of the problem. It is composed of 68 duties. During the optimization process, the number of duties are decreased, the efficiency is increased and the objective function value is decreased, as expected.

Table 2. Clustered-SEBACS model solutions for different nodes of the branch-and-cut tree.

Node number	Duties	Efficiency	Objective function
1	145	0.212	272.170
5	102	0.319	183.616
10	79	0.491	133.272
15	69	0.665	110.464
20	68	0.691	108.600

Regarding computational times, the greatest effort was done for the Lagrangian relaxation, where the convergence criterion was reached after 725 seconds. However, computational times for the RINS to get the optimum were approximately than 5 seconds. Note that 5 seconds are needed to reach node number 20 in Figure 1.

7. Conclusions

We have proposed a new approach to solve the crew scheduling problem. The main novelty is that it provides a new point of view about it which allows its integration with other rail planning problems.

We show that a sequencing-based formulation may be appropriate for the crew scheduling problem in the context of rapid transit networks. We solve the problem using a heuristic, the relaxation induced neighborhood search, and demonstrate that it provides optimal solutions by computing lower bounds using the Lagrangian relaxation.

Preliminary computational experiments drawn from RENFE, the major railway operator in Spain, show that computational times needed in order to find optimal solutions are low enough to even use this approach in real-time applications such as those related to disruption management.

In our future research we are going to embark on the integration of the crew scheduling problem with other railway planning problems, such as rolling stock circulations, and in the development of a new approach for the crew management during disruptions.

Acknowledgements

This research was supported by Project Grants TRA2011-27791-C03-01 and TRA2014-52530-C3-1-P by the "Ministerio de Economía y Competitividad, Spain".

References

Abbink, E., Fischetti, M., Kroon, L., Timmer, G., Vromans, M. (2005). Reinventing crew scheduling at Netherlands Railways. Interfaces, 35(5), 393-401.

Barnhart, C., Cohn, A. M., Johnson, E. L., Klabjan, D., Nemhauser, G. L., Vance, P. H. (2003). Airline crew scheduling. In Handbook of transportation science (pp. 517-560). Springer US.

Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W., Vance, P. H. (1998). Branch-and-price: Column generation for solving huge integer programs. Operations research, 46(3), 316-329.

Bomdörfer, R. (1998). Aspects of set packing, partitioning, and covering. Doctoral dissertation, Ph. D. thesis, Technischen Universitat Berlin, Berlin, Germany.

Cadarso, L., Marín, Á. (2010). Robust routing of rapid transit rolling stock. Public Transport, 2(1-2), 51-68.

Cadarso, L., Marín, Á. (2011). Robust rolling stock in rapid transit networks. Computers & Operations Research, 38(8), 1131-1142.

Cadarso, L., Marín, Á. (2014). Improving robustness of rolling stock circulations in rapid transit networks. Computers & Operations Research, 51, 146-159.

Caprara, A., Fischetti, M., Toth, P., Vigo, D., Guida, P. L. (1997). Algorithms for railway crew management. Mathematical programming, 79(1-3), 125-141.

Danna, E., Rothberg, E., Le Pape, C. (2005). Exploring relaxation induced neighborhoods to improve MIP solutions. Mathematical Programming, 102(1), 71-90.

Fisher, M. L. (2004). The Lagrangian relaxation method for solving integer programming problems. Management science, 50, 1861-1871.

Huisman, D., Kroon, L.G., Lentink, R.M., Vromans, M.J.C.M. (2005). Operations Research in passenger railway transportation. Statistica Neerlandica , 59(4), 467-497.

Kroon, L., Fischetti, M. (2001). Crew scheduling for Netherlands railways "destination: customer". In Computer-aided scheduling of public transport (pp. 181-201). Springer Berlin Heidelberg.

Kumar, A., Vaidyanathan, B., Jha, K.C., Ahuja, R.K. (2009). Railroad Crew Scheduling. In Encyclopedia of Optimization (pp. 3227-3236). Springer.

Lemaréchal, C. (2001). Lagrangian relaxation. In Computational combinatorial optimization (pp. 112-156). Springer.

Van den Bergh, J., Beliën, J., De Bruecker, P., Demeulemeester, E., De Boeck, L. (2013). Personnel scheduling: A literature review. European Journal of Operational Research, 226(3), 367-385.